



Privacy Preserving Data Analytics for Smart Homes

Nisha Gadge, Shruti Rokade, Priyanka More

Information Technology Department
G. S. M. College of Engineering
India

Abstract— *The Cloud computing has emerged as one of the most influential paradigms in the IT industry in latter years. Since this is new computing technology requires users to entrust their valuable data to cloud service providers, there are increasing security and confidentiality concerns on the data which is utilized. Several schemes using attribute-based encryption (ABE) have been proposed for access control of outsourced data in cloud computing. Most of these which are discomfort from inflexibility in implementing complex access control policies. In order to realize scalable, fine grained access control and the scalable outsourcing of the data in cloud computing, in our paper, we are going to appliance hierarchical attribute-set-based encryption by extending cipher text policy attribute-set-based encryption with a hierarchical structure of users. The implemented system achieves scribbled due to its hierarchical structure and inherits fine-grained access control and flexibility in supporting compound attributes of attribute set based encryption. In addition, the hierarchical aspect set based encryption helps in multiple value assignments for access expiration time to deal with user revocation more efficiently than actual schemes. We implement our scheme and show that it is both efficient and flexible in dealing with access control for outsourced data in cloud computing with comprehensive experiments.*

Keywords: ABSE, KP-ABE, Owner module, Consumer module, Server module, Encryption module.

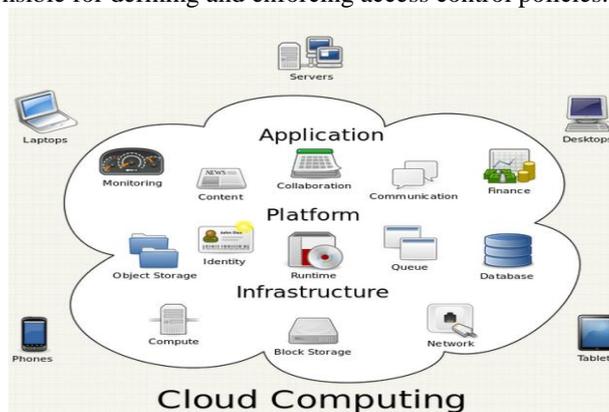
I. INTRODUCTION

Cloud is nothing but the sharing of data among the multiple users [1][4]. The cloud computing service provides the business applications through the web browser to the user. In this the data and the software are placed over the virtual server. This server is also known as the cloud server.

The service provider is responsible for providing the service related to the requested data to the user. Cloud helps to upload data in very big size which can be shared among the users. Cloud computing has emerged as one of the most useful technique in information technology [4]. Cloud computing is an umbrella term used to refer the internet based development and services. Cloud computing has some characteristics like remotely hosted, co modified [8]. In these types of systems sometimes data may be placed on the different infrastructure.

We observe that there are also cases in which cloud users themselves are content providers. They publish data on cloud servers for sharing and need fine-grained data access control in terms of which user (data consumer) has the access privilege to which types of data. The data owners want to keep data on cloud very confidential and they also want to take the maximum advantage of the resources that cloud provides. The data access control has been evolving in the past thirty years and various techniques have been developed to effectively implement fine grained access control.

These techniques will allow flexibility in accessing data from different users in cloud environment. The available access control architectures usually assume the data owner and the servers which contains the same domain. Here the servers are fully entrusted responsible for defining and enforcing access control policies.



The various advantages of cloud computing include reduced costs and increased operational flexibility and scalability and so on [16].

The cloud platform allows the developers to create the applications that will run in the cloud. The cloud storage is the concept in which the data is placed on the cloud server instead of user's local drive. The infrastructure provides the user different services which reduces the cost of maintaining the software. The hardware can be upgraded in this kind of the infrastructure provided by the cloud. The cloud services helps in cloud to connect the different users through network.

II. SYSTEM ANALYSIS

A. EXISTING SYSTEM

In current systems the user has the restrictions about sharing the data of other data owner. The user can only have the authority of accessing the data which lies where he has registered. The data which is encrypted when requested by user needs to go through the data owner for decryption. When fine grained access control their is heavy computation overhead on the data owner for key distribution and data management. These systems do not scale well .Cloud providers should provide privacy about the data which is saved by the owner. So the security issue matters lot. Now days user need data to be available very quickly so the single cloud is unable to provide such a fast access of the data to the users. It is not sure that when the data is deleted from cloud that data remains in cloud or not. So these kinds of issues are in the current system.

B. PROPOSED SYSTEM

The proposed system overcomes the disadvantages of the current system by providing the flexible access and secure data [16]. It defines and enforces access policies based on attributes of the data [17]. The data owner is able to delegate most of the computation tasks involved in fine grained data access control to un-trusted cloud servers without disclosing the underlying contents of the data. This is a hierarchical attribute-set-based encryption scheme for access control in cloud. It is the extension of the ciphertext policy attribute set based encryption scheme with a hierarchical structure of system users. It also achieves scalable, flexible and fine-grained access control [16].In this system we associate each data file with a set of attributes, and assign each user an separate access structure which is defined under the different data owners. These data owners are under the one domain authority which is the part of the cloud service. To achieve this type of access control, we used the technique called key policy attribute based encryption to get the keys over the data which is in encrypted format [17]. It helps to get the fine graininess of access control.

III. MODULES

A. DATA OWNER MODULE

The data owner can upload the data on the cloud in this module. In this system data owner has the authority to get the sub owners which can directly put their data on the cloud through the owner module [2]. They are the sub domains of the data owner module. In this module the owner can modify the data which is placed on the cloud [3]. The data owner is able to see the history of its data. The main advantage of this module is module owner is able to retrieve the data which is deleted in authorized or in unauthorized way.

This module has the option of retrieving that data. In this module the data owner can update the data and also he is able to update the expiration time. The owner can manipulate the encrypted data file and the owner can set the access privilege to the data files which are encrypted. The owner creates the username and password after the registration [5]. This password can be changed afterword.

B. DATA CONSUMER MODULE

In this module authority is given to the new consumer and registered consumer. The username and password is generated in this module by using that username and password consumer can access its data in future. In this module consumer is able to get the data which he wants. Consumer request for the data then he gets the key through email [12]. By using this key consumer is able to download the data in decrypted format which is already encrypted on the cloud [14]. Consumer has the restriction about updating data. Consumer has only rights to share the data.

For the user level, all the privileges are given by the data owner within the particular domain and the Data users are controlled by the higher level authority only that is the owner [13]. User wants to access data files either within the domain or outside the scope of its domain, so malicious users may collude with each other to get sensitive files beyond their privileges. In this module when consumer wants to download any data which is encrypted he needs the data to be decrypt to read the downloaded data. This needs the decryption key [15]. This key is provided by the system through the email of the consumer. Every time when he downloads the data that time it needs to be decrypt to read this so the key is provided which is new at every time.

C. DATA SERVER MODULE

This module is nothing but the server where the data is placed. This module manages the data which is placed on the same server. This data is in encrypted format. In this module the data owners puts its data on the cloud in the encrypted format and the user can share the data form this module [5]. To share the files from this module consumer has to decrypt this data and then user can get that data.

D. DATA ENCRYPTION MODULE

In this module the keys are generated like master, public and private keys. The trusted domain authority is responsible for generating and distributing system parameters and root master keys [11]. A domain authority is responsible for

delegating keys to subordinate domain authorities at the next level. Every user in the system is assigned a key structure which specifies the attributes associated with the user's decryption key [12]. With the help of algorithm the system public parameters PK and master key MK are generated. PK will be made public to other parties and MK maintained confidential [13]. A user, when sends request for data files stored on the cloud server, the cloud server sends the respective ciphertext to the user [17][16]. By calling decrypt (CT, SK) to obtain DEK and then decrypt data files using DEK the user decrypts the data

IV. TECHNIQUES USED

A. Key Policy Attribute-Based Encryption (KP-ABE)

KP-ABE [6] is a public key cryptography primitive for one-to-many conversation. In KP-ABE, data is associated with attributes for each of which a public key fundamental is defined. The set of attributes to the message by encrypting it with the corresponding public key components associated with the encryptor. A KP-ABE scheme is easygoing of four algorithms which can be defined as follows:

Setup: This algorithm takes as input a security parameter κ and the attribute universe $U = \{1, 2, \dots, N\}$ of cardinality N . It defines a bilinear group G_1 of prime order p with a generator g , a bilinear map $e: G_1 \times G_1 \rightarrow G_2$ which has the properties of blinder, comparability, and non-degeneracy. It restitution the public key as well as a system master key MK as follows $PK = (Y, T_1, T_2, \dots, T_N)$ $MK = (y, t_1, t_2, \dots, t_N)$ where $T_i \in G_1$ and $t_i \in \mathbb{Z}_p$ are for attribute i , $1 \leq i \leq N$, and $Y \in G_2$ is another public key component. We have $T_i = gt_i$ and $Y = e(g, g)y$, $y \in \mathbb{Z}_p$. While PK is publicly known to all the parties in the system, master key is kept as a secret by the authorisation party. Encryption: This algorithm takes a message M , a set of characteristics I as input and public key PK, It outputs the cipher text E with the following format: $E = (I, \tilde{E}, \{E_i\}_{i \in I})$ where $\tilde{E} = MY^s$, $E_i = T_{s_i}$, and s is randomly chosen from \mathbb{Z}_p . Key Generation: This algorithm takes as input an access tree T , the master key MK, and the public key PK. It gives a user secret key SK as follows. It defines a random polynomial $p_i(x)$ for each node i of T in the top-down manner starting from the root node r . For each non-root node j , $p_j(0) = p_{\text{parent}(j)}(\text{idx}(j))$ where $\text{parent}(j)$ represents j 's parent and $\text{idx}(j)$ is j 's unique index given by its parent. For the root node r , $p_r(0) = y$. Then it results SK as follows. $SK = \{sk_i\}_{i \in L}$ where L denotes the set of attributes attached to the leaf nodes of T and $sk_i = p_i(0) t_i$. Decryption: This algorithm takes as input the ciphertext E encrypted under the attribute set I which is the user's the public key PK and secret key SK for access tree T . First it computes $(E_i, sk_i) = e(g, g)p_i(0)s$ for leaf nodes. It sum these pairing results in the bottom-up manner using the polynomial interpolation technique. At the end it may recover the blind factor $Y^s = e(g, g)y^s$ and output the message M if and only if I satisfies T .

B. Proxy Re-Encryption (PRE)

Proxy Re-Encryption (PRE) is a cryptographic primitive in which a semi-trusted proxy is able to convert a cipher text encrypted under Alice's public key into another cipher text that can be opened by Bob's private key without seeing the underlying plaintext.

V. SECURITY ANALYSIS

The security analysis of the system is respecting of the following few factors

A. Fine Grained Access Control:

The data owner is able to define and enforce expressive and flexible access structure for each user. The access structure of each user is defined as a logic formula over data file attributes, which is able to represent any in demand data file set.

B. User Secrete Key Accountability:

This property can be immediately achieved by using the enhanced construction of KP-ABE which can be used to disclose the identities of key abusers. We analyze data confidence of the system by giving a cryptographic security proof.

C. Data Confidentiality:

We analyze data confidentiality of the system by comparing it with an intuitive scheme in which data files are encrypted using symmetric DEKs. These DEKs are encrypted straight using standard KP-ABE. In the system just cipher text of files are given to the cloud servers. The basic KP-ABE is provably insure under the attribute-based Selective-Set model [6] given the Decisional Bilinear Diffie-Hellman (DBDH) problem is hard.

VI. CONCLUSION

In this system, to realize flexible, fine grained and scalable access control in cloud we have implemented extension over the hierarchical attribute set based encryption scheme. This scheme seamlessly incorporates a hierarchical structure of system users by applying a delegation algorithm to Attribute set based encryption. This system not only supports compound attributes, but also achieves efficient user revocation due to multiple value assignments of attributes. We formally proved the security of this system based on the security of key policy attribute based encryption. Finally, we implemented the proposed scheme, and conducted better performance analysis, which showed its efficiency and advantages over existing schemes. In future we can achieve more security by providing the generated key in different ways.

REFERENCES

CLOUD COMPUTING, IEEE INFOCOM, 2010.

- [1] R. Buyya, C. ShinYeo, J. Broberg, and I. Brandic, —Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility,| *Future Generation Comput. Syst.*, vol. 25, pp. 599–616, 2009
- [2] Dr. Suhas H. Patil, —Data Security technique in cloud storage|, *International Journal of Computer Science and Technology(IJCST)*, Vol.4, Issue,2-3.
- [3] Amazon Web Services (AWS) [Online]. Available: <https://s3.amazonaws.com/>
- [4] R. Martin, —IBM brings cloud computing to earth with massive new data centers,| *InformationWeek Aug. 2008* [Online]. Available: http://www.informationweek.com/news/hardware/data_centers/209901523
- [5] Google App Engine [Online]. Available: <http://code.google.com/appengine/>
- [6] K. Barlow and J. Lane, —Like technology from an advanced alien culture: Google apps for education at ASU,| in *Proc. ACM SIGUCCS User Services Conf.*, Orlando, FL, 2007.
- [7] D. E. Bell and L. J. LaPadula, *Secure Computer Systems: Unified Exposition and Multics Interpretation* The MITRE Corporation, Tech. Rep., 1976.
- [8] H. Harney, A. Colgrove, and P. D. McDaniel, —Principles of policy in secure groups,| in *Proc. NDSS*, San Diego, CA, 2001.
- [9] P. D. McDaniel and A. Prakash, —Methods and limitations of security policy reconciliation,| in *Proc. IEEE Symp. Security and Privacy*, Berkeley, CA, 2002.
- [10] J. Li, N. Li, and W. H. Winsborough, —Automated trust negotiation using cryptographic credentials,| in *Proc. ACM Conf. Computer and Communications Security (CCS)*, Alexandria, VA, 2005.
- [11] S. Yu, C. Wang, K. Ren, and W. Lou, —Achieving secure, scalable, and fine-grained data access control in cloud computing,| in *Proc. IEEE INFOCOM 2010*, 2010, pp. 534–542
- [12] H. Harney, A. Colgrove, and P. D. McDaniel, “Principles of policy in secure groups,” in *Proc. of NDSS’01*, 2001.
- [13] W. H. Winsborough and J. Li, N. Li, , “Automated trust negotiation using cryptographic credentials,” in *Proc. of CCS’05*, 2005.
- [14] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, “Scalable secure file sharing on untrusted storage,” in *Proc. Of FAST’03*,
- [15] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, “Improved proxy re-encryption schemes with applications to secure distributed storage,” in *Proc. of NDSS’05*, 2005.
- [16] Shucheng Yu, Cong Wang, Kui Ren , and Wenjing Lou. ” Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing”, in *IEEE INFOCOM* ,2010.
- [17] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling public verifiability and data dynamics for storage security in cloud computing,” in *Proc. of ESORICS ’09*, 2009.